CLAIMS

What is claimed is:

1. A method for a device to download data from a data source while connected to a host, comprising:

connecting to a host;

waiting for a request signal from the host;

responding to the request signal with a negative acknowledgement (NAK) to intentionally postpone a response to the request signal; and

downloading data from the data source for a predetermined time period based on the request signal.

2. The method of claim 1, the data source comprising non-volatile memory.

3. The method of claim 2, the non-volatile memory is at least one of an Electrically Erasable Programmable Read Only Memory (EEPROM) and a flash memory.

4. The method of claim 2, further comprising reading a signature from the non-volatile memory and validating the signature prior to connecting to the host.

5. The method of claim 2, further comprising reading descriptor information from the non-volatile memory prior to connecting to the host.

6. The method of claim 1, further comprising setting a pointer for tracking data downloaded from the data source.

7. The method of claim 1, further comprising repeating waiting for a request signal from the host, responding to the request signal with a NAK and

downloading data from the data source for a predetermined time period based on the request signal, until the data downloaded from the data source has completed.

8. The method of claim 7, further comprising updating a download pointer each time the predetermine time period is completed.

9. The method of claim 8, the predetermined time period is a first time period for a data request signal, and a second time period for a status request signal.

10. The method of claim 9, the first time period is about five hundred milliseconds and the second time period is about fifty milliseconds.

11. The method of claim 1, the host is a USB host and the device is a USB device.

12. The method of claim 1, the predetermined time period is monitored by a timer.

13. The method of claim 1, further comprising determining and downloading a number of data blocks to be downloaded based on the predetermined time period.

14. The method of claim 13, the number of data blocks to be downloaded being further based on at least one of a download data rate and a block size.

15. The method of claim 13, further comprising setting a loop counter based on the number of data blocks to be downloaded.

16. A method for a Universal Serial Bus (USB) device to download firmware while connected to a USB host, comprising

connecting to a USB host;

waiting for a request signal from the USB host;

downloading data blocks associated with firmware from a data source based on a predetermined time period associated with the request signal type; and

repeating the waiting for a request signal and the downloading data blocks from the data source, until the downloading of the firmware is complete.

17. The method of claim 16, further comprising updating a download pointer that tracks the last data block downloaded.

18. The method of claim 16, further comprising sending a negative acknowledgement (NAK) to the host to intentionally postpone the transmission of the response to the request signal.

19. The method of claim 16, further comprising determining the signal request type and setting the predetermined time period accordingly.

20. The method of claim 16, further comprising determining a number of data blocks to be downloaded based on the predetermined time period.

21. The method of claim 20, the number of data blocks to be downloaded being further based on at least one of a download data rate and a block size.

22. A universal serial bus (USB) compatible device, comprising:

a non-volatile memory having firmware stored therein; and

a microcontroller unit (MCU) that upon reset configures the device, such that the device responds with a negative acknowledgement (NAK) in response to a request signal from a host controller, until downloading of firmware to the MCU has completed.

23. The device of claim 22, further comprising a memory for storing a download pointer to track the firmware download.

24. The device of claim 22, the MCU downloads data blocks associated with the firmware for a predetermined time period based on the request signal type from the host controller.

25. The device of claim 24, the MCU further determines the number of data blocks to be downloaded for the predetermined time period based on download data rate and a block size.

26. The device of claim 24, the predetermined time period is a first time period for a signal with a data stage and a second time period for a signal without a data stage.

27. The device of claim 26, the first time period is about five hundred milliseconds and the second time period is about fifty milliseconds.

28. The device of claim 23, further comprising a timer for monitoring the firmware download.